IRELL & MANELLA LLP
Morgan Chu (SBN 70446)
(mchu@irell.com)
Jonathan S. Kagan (SBN 166039)
(jkagan@irell.com)
1800 Avenue of the Stars, Suite 900
Los Angeles, California 90067-4276
Telephone: (310) 277-1010
Facsimile: (310) 203-7199

David C. McPhie (SBN 231520)
(dmcphie@irell.com)
Douglas J. Dixon (SBN 275389)
(ddixon@irell.com)
Nima Hefazi (SBN 272816)
(nhefazi@irell.com)
840 Newport Center Drive, Suite 400
Newport Beach, California 92660-6324
Telephone: (949) 760-0991
Facsimile: (949) 760-5200

Attorneys for Defendant
JUNIPER NETWORKS, INC.

UNITED STATES DISTRICT COURT
FOR THE NORTHERN DISTRICT OF CALIFORNIA
SAN FRANCISCO DIVISION

IMPLICIT NETWORKS, INC.,

Plaintiff,

v.

JUNIPER NETWORKS, INC.,

Defendant.

Case No. C 10-4234 SI

JUNIPER NETWORKS, INC.'S NOTICE OF MOTION AND MEMORANDUM OF LAW IN SUPPORT OF MOTION FOR SUMMARY JUDGMENT OR, IN THE ALTERNATIVE, FOR PARTIAL SUMMARY JUDGMENT, ON THE ISSUE OF INVALIDITY

Date: December 14, 2012
Time: 9:00 a.m.
Courtroom: 10

NOTICE OF MOTION

PLEASE TAKE NOTICE that on Friday, December 14, 2012, at 9:00 a.m., or as soon thereafter as counsel may be heard, pursuant to Rule 56 of the Federal Rules of Civil Procedure, defendant Juniper Networks, Inc. ("Juniper") will, and hereby does, move for an order granting summary judgment of invalidity of claims 1, 15, and 35 of U.S. Patent No. 6,629,163 ("the '163 patent") and claims 1, 4, and 10 of U.S. Patent No. 7,711,857 ("the '857 patent") (collectively, the "asserted claims") or, in the alternative, partial summary judgment with respect to selected elements of those claims. Juniper's motion is based upon this Notice, the Memorandum of Points and Authorities below, the supporting Expert Declaration of Dr. Kenneth L. Calvert ("Calvert Declaration") and the supporting documents attached thereto, the Declaration of Douglas J. Dixon in Support of Juniper Networks, Inc.'s Motion for Summary Judgment, or, in the Alternative, Partial Summary Judgment, on the Issue of Invalidity, and the supporting documents attached thereto, and Exhibits 26 and 28 to the Declaration of Nima Hefazi in support of Juniper Networks, Inc.'s Motion for Summary Judgment of Non-Infringement.

Dated: November 12, 2012

Respectfully submitted,

IRELL & MANELLA LLP

By: */s/ Douglas J. Dixon*

David C. McPhie (dmcphie@irell.com)
Douglas J. Dixon (ddixon@irell.com)
840 Newport Center Drive, Suite 400
Newport Beach, California 92660-6324
Telephone: (949) 760-0991
Facsimile: (949) 760-5200

Morgan Chu (mchu@irell.com)
Jonathan S. Kagan (jkagan@irell.com)
1800 Avenue of the Stars, Suite 900
Los Angeles, California 90067-4276
Telephone: (310) 277-1010
Facsimile: (310) 203-7199

Attorneys for Defendant Juniper Networks, Inc.

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Page

Cases

Statutes

Other Authorities

STATEMENT OF THE ISSUES TO BE DECIDED

This motion seeks to dispose of all the asserted claims in this case – claims 1, 15, and 35 of U.S. Patent No. 6,629,163 ("the '163 patent") and claims 1, 4, and 10 of U.S. Patent No. 7,711,857 ("the '857 patent") (collectively, the "asserted claims"). As explained below, Juniper seeks an order of summary judgment that the asserted claims are invalid as:

(a) anticipated or rendered obvious by Daniel Decasper, *et al.*, "Router Plugins: A Software Architecture for next Generation Routers," Computer Communication Review, a publication of ACM SIGCOMM, Vol. 28, No. 4, Oct. 1998 (hereinafter, "**Decasper98**") (Dixon Ex. 1) (*see* Part IV below); or

(b) rendered obvious by Decasper98 in combination with IBM, Local Area Network Products Concepts and Products: Routers and Gateways (May 1996) (hereinafter, "**IBM96**") (Dixon Ex. 2) and Mark Nelson and Jean-Loup Gailly, *The Data Compression Book*, M&T Books (2nd ed. 1996) (hereinafter, "**Nelson**") (Dixon Ex. 3) (*see* Part V below).

In the alternative, Juniper seeks an order of partial summary judgment that specified limitations of the asserted claims are (a) anticipated or rendered obvious by Decasper98 or (b) rendered obvious by Decasper98 in view of IBM96 and Nelson. (*See* Part VII, below.)

The invalidity analysis for this case is straightforward. Decasper98 and the other cited references were all published more than a year before the filing date of the patents (December 29, 1999). Implicit does not contest the status of the cited references as prior art publications under the statutory bar of 35 U.S.C. § 102(b). There are no priority battles or "first-to-invent" disputes to resolve. The only remaining question is whether Decasper98 and the other references disclose or render obvious each of the three disputed elements of the asserted claims, from the point of view of one of ordinary skill in the relevant technical art.

On this last question, the technically trained examiners of the United States Patent and Trademark Office ("PTO") have provided some significant assistance: the prior art references and invalidity theories presented in this motion have already been carefully considered by two panels of examiners in the pending *inter partes* reexamination proceedings for the patents-in-suit. Those examiners have unanimously concluded that all asserted claims of the patents are invalid over the Decasper98 reference alone (among other grounds). This conclusion was most recently confirmed

in a detailed, 140-page "Action Closing Prosecution" that considered – and rejected – the very same validity arguments that Implicit's expert now advances in this litigation.

Although the PTO applies a somewhat different standard for evaluating validity than that employed in litigation, there is nothing about the way that the reexamination proceedings have unfolded that suggests this is a close case. Indeed, the Federal Circuit has recognized that the burden of proving invalidity may be more easily carried where (as here) none of the references at issue were ever considered during any of the prior prosecution proceedings relating to the patents.

In short, Decasper98 and others publicly disclosed the claimed invention well over a year before Implicit even filed its application for the patents. Accordingly, Juniper submits that the Court should grant summary judgment of invalidity for the reasons set forth in detail below.

STATEMENT OF THE RELEVANT FACTS

A. The Patents Claim A Method Of Processing Packets Of A Message By "Dynamically Identifying . . . [A] Sequence Of Components."

The primary method today of transmitting messages (*e.g.*, picture, email, video, etc.) over a computer network (*e.g.*, the Internet) is to break down the message into discrete blocks of data known as packets, which are then reassembled into the original message at the destination. (*See* Calvert Ex. 2[1] (Calvert Invalidity Report) ¶ 77.) Implicit claims that its patents cover a particular method of handling multi-packet messages. (Dixon Ex. 4[2] ('163 Patent) at 1:11-12, 2:57-64[3].) The patents claim an approach to processing packets of data that starts by analyzing the first packet of a message in order to figure out how best to process the remaining packets in the message, including conversion, if necessary, from one format to another. (*See* Dixon Ex. 4 at 2:38-3:9.) This packet processing involves multiple steps, and it is therefore necessary to select and use multiple software routines or "components" – one component for each step – like modular

[1] "Calvert Ex. __" refers to the exhibits to the Declaration of Kenneth L. Calvert, Ph.D., executed November 9, 2012.

[2] "Dixon Ex. __" refers to the exhibits to the Declaration of Douglas J. Dixon in Support of Juniper Networks, Inc.'s Motion for Summary Judgment, or, in the Alternative, Partial Summary Judgment, on the Issue of Invalidity, executed November 9, 2012.

[3] The '163 patent and the '857 patent share a common specification. For ease of reference, Juniper refers to the '163 specification throughout this brief unless noted otherwise.

stations on an assembly line. (*Id.* at 2:47-49.) One of the key aspects of the claims is that this sequence of components is created or formed only "*after the first packet is received*" – *i.e.*, it must be a *dynamically identified sequence of components*. (*See* Dixon Ex. 5 ('163 patent C1) at 1:35-43.) This dynamically identified sequence is then stored so it can be used in processing each subsequent packet of the same message. (*See id.* at 1:40-43, 54-56.) The purported goal of Implicit's system is to avoid "[t]he overhead of statically providing each possible series of conversion routines" in advance, by instead "dynamically identifying" and selecting individual components for an appropriate sequence of software routines only ***after*** the first packet of the message is received. (*See* Dixon Ex. 4 at 1:38-2:5.)

B. The '163 Patent Escapes The *Ex Parte* Reexamination Only By Amending The Claim Language.

Implicit filed its application for the '163 patent on December 29, 1999, and the patent subsequently issued on September 30, 2003. It was several years after this that Implicit's software business collapsed and Implicit began asserting its patent portfolio against companies as a means for generating revenue. One of Implicit's litigation targets elected to file an *ex parte* reexamination request for the '163 patent, and on January 17, 2009, the PTO granted the request and initiated reexamination proceedings. The PTO quickly determined that all of the original claims of the '163 patent were invalid as anticipated. (Dixon Ex. 7 (7/7/2009 Office Action) at 5-13.) In an attempt to overcome these rejections, Implicit emphasized to the PTO that the purported invention included a critical timing aspect: "***first***, packet is received, and ***then***, component sequence is identified based on packet." (Dixon Ex. 8 (Examiner Interview PowerPoint) at 2.) Implicit pointed to language from the specification suggesting a "dynamic" approach would avoid the "overhead" that would otherwise be involved in calculating "each possible series of conversion routines" in advance. (Dixon Ex. 4 at 1:38-66.) Implicit asserted that the prior art, by contrast, performed its identification of sequences before the first packet was received, and therefore did not disclose the type of dynamic identification with the associated reduction in overhead contemplated by the claims. After multiple rejections, Implicit ultimately amended its claims to expressly include the steps of "dynamically identifying" and "selecting

individual components". (*See, e.g.*, Dixon Ex. 9 (12/18/2009 Response to Final Rejection) at 10; Dixon Ex. 10 (2/8/2010 Amendment After Final) at 12.) Thereafter, the examiners issued a notice of allowance for the asserted claims of the '163 patent as amended based expressly on Implicit's argument that the prior art "does not dynamically identify sequences". (Dixon Ex. 11 (3/2/2010 Notice of Intent to Issue *Ex Parte* Reexamination Certificate) at 4.) The PTO issued the reexamination certificate on June 22, 2010. (Dixon Ex. 5.)

The '857 patent, which is a continuation of and claims priority to the '163 patent, issued on May 4, 2010. (*See* Dixon Ex. 6 ('857 patent).)

C. The PTO Rejects The Asserted Claims As Invalid In Concurrent *Inter Partes* Reexamination Proceedings.

Juniper is a successful networking technology company that happened to be one of the latest set of companies to be sued as part of Implicit's ongoing litigation campaign. As Juniper began to learn about Implicit's broad infringement contentions and claim construction positions, it identified a number of prior art references – including Decasper98 – that appeared to closely match the asserted claims but had never before been disclosed or considered in the prior prosecution and *ex parte* reexamination proceedings. Accordingly, Juniper filed requests for *inter partes* reexamination of both patents in early 2012. Soon thereafter, the PTO rejected every single asserted claim of the patents on one or more of the following grounds:

- Anticipated by or obvious in light of Decasper98;
- Obvious in view of Decasper98 in combination with IBM96 and (optionally) Nelson;
- Anticipated by or obvious in light of the U.S. Patent No. 6,243,667 to Kerr, *et al.*, issued on June 5, 2001 ("Kerr"); and
- Obvious in view of a 1996 article by Tom Pfeifer in combination with Kerr.

Implicit submitted a written response to the PTO's rejections, which the PTO duly considered and rejected[4], confirming the invalidity of every single asserted claim of the '163 patent on multiple grounds. (*See generally* Dixon Ex. 15 ('163 Patent Action Closing

[4] So far the PTO has only rejected Implicit's arguments in the '163 reexamination; an Action Closing Prosecution has not yet issued in the '857 reexamination. Nevertheless, Juniper has every reason to expect that a similar result will hold in the '857 reexamination.

Prosecution).) ***In so doing, the PTO rejected each and every one of the same validity arguments that Implicit and its expert have advanced in this litigation***. Juniper now asks this Court to similarly reject the asserted claims as invalid on the same grounds already accepted by the PTO.

ARGUMENT

I. LEGAL PRINCIPLES

A. The Summary Judgment Standard.

"Summary judgment is appropriate when there is no genuine issue of material fact and the moving party is entitled to judgment as a matter of law." *Bristol-Myers Squibb Co. v. Ben Venue Labs., Inc.*, 246 F.3d 1368, 1374 (Fed. Cir. 2001). Although patents are presumed valid (35 U.S.C. § 282), they can also be invalidated with the kind of "clear and convincing" evidence present in this case. *See Telemac Cellular Corp. v. Topp Telecom, Inc.*, 247 F.3d 1316, 1326 (Fed. Cir. 2001) (affirming summary judgment of invalidity by anticipation); *Ryko Mfg. Co. v. Nu-Star, Inc.*, 950 F.2d 714, 716 (Fed. Cir. 1991) ("a district court can properly grant, as a matter of law, a motion for summary judgment on patent invalidity"). Significantly, "the alleged infringer's burden may be more easily carried" where (as here) validity is challenged on the basis of prior art "that was not before the examiner" during the original prosecution proceedings. *SIBIA Neurosciences, Inc. v. Cadus Pharm. Corp*., 225 F.3d 1349, 1355-56 (Fed. Cir. 2000).[5]

There is a strong policy favoring the dismantling of improper patent monopolies, and for this reason "[t]he validity of a patent is always subject to plenary challenge on its merits." *Magnivision, Inc. v. Bonneau Co.*, 115 F.3d 956, 960 (Fed. Cir. 1997). As the Supreme Court has recognized: "[T]he results of ordinary innovation are not the subject of exclusive rights under the patent laws. Were it otherwise patents might stifle, rather than promote, the progress of useful arts." *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 427 (2007).

Once "evidence is presented establishing a prima-facie case of [patent] invalidity, the opponent of invalidity must come forward with evidence to counter the prima facie challenge

[5] Although the PTO applies a lower "preponderance of the evidence" standard in evaluating invalidity during reexamination proceedings, the prior art at issue in this motion unmistakably satisfies the clear and convincing standard as well – especially since the PTO has never before considered invalidity in light of Decasper98 or the other references cited here.

. . . ." *Cable Elec. Prods., Inc. v. Genmark, Inc.*, 770 F.2d 1015, 1022 (Fed. Cir. 1985), *overruled on other grounds by Midwest Indus., Inc. v. Karavan Trailers, Inc.*, 175 F.3d 1356, 1358-60 (Fed. Cir. 1999). Not just any counter-evidence will overcome a prima facie case of invalidity. As the Supreme Court stated in *Anderson v. Liberty Lobby, Inc.*, 477 U.S. 242, 247-48 (1986), "the mere existence of *some* alleged factual dispute between the parties will not defeat an otherwise properly supported motion for summary judgment; the requirement is that there be no *genuine* issue of *material* fact." *See also Paragon Podiatry Lab., Inc. v. KLM Lab., Inc.*, 984 F.2d 1182, 1184-85 (Fed. Cir. 1993) (affirming summary judgment of anticipation). A dispute involving a material fact is genuine only "if the evidence is such that a reasonable jury could return a verdict for the nonmoving party". *Anderson*, 477 U.S. at 248.

General assertions of facts, general denials, and conclusory statements are insufficient to shoulder the non-movant's burden. *Johnston v. IVAC Corp.*, 885 F.2d 1574, 1578 (Fed. Cir. 1989). And mere denials or conclusory statements are insufficient to create an evidentiary conflict. *Collins, Inc. v. N. Telecom, Ltd.*, 216 F.3d 1042, 1046 (Fed. Cir. 2000).

B. A Patent Claim Is Invalid If Anticipated.

To qualify for patent protection, an invention must be novel. An invention is not novel if it has been "anticipated" by the prior art. Section 102 describes several categories of anticipatory prior art. One of the categories outlined in § 102(b) is as follows: a person is not entitled to a patent if his or her invention "was patented or described in a printed publication in this or a foreign country . . . more than one year prior to the date of the application for patent in the United States." Section 102(b) acts as an absolute bar. *In re Bayer*, 568 F.2d 1357, 1359-60 (C.C.P.A. 1978).[6]

The classic test for determining whether the subject matter recited in a patent claim is found in a prior art reference is: "That which infringes, if later, would anticipate, if earlier." *Peters v. Active Mfg. Co.*, 129 U.S. 530, 537 (1889); *see also Bristol Myers*, 246 F.3d at 1378 ("[I]t is axiomatic that that which would literally infringe if later anticipates if earlier."). In modern parlance, "[t]o anticipate a claim, a prior art reference must disclose every limitation of

[6] *See also* 35 U.S.C. § 102(a) (a person is not entitled to a patent if his or her invention was described in a printed publication "before the invention" claimed in the patent application).

the claimed invention, either explicitly or inherently." *Mehl/Biophile Int'l Corp. v. Milgraum*, 192 F.3d 1362, 1365 (Fed. Cir. 1999). But a prior art reference need not recite the claim's limitations word-for-word. "[A] prior art reference may anticipate when the claim limitation or limitations not expressly found in that reference are nonetheless inherent in it." *Id.* "[I]f the prior art necessarily functions in accordance with, or includes, the claimed limitations, it anticipates." *Id.* Moreover, "[w]hen the reference is silent about the asserted inherent characteristic, such gap in the reference may be filled with recourse to extrinsic evidence" without affecting the anticipatory nature of the reference. *Continental Can Co. USA v. Monsanto Co.*, 948 F.2d 1264, 1268 (Fed. Cir. 1991). "This modest flexibility in the rule . . . accommodates situations in which the common knowledge of the technologists is not recorded in the reference." *Id.*

C. A Patent Claim Is Invalid If Obvious.

In addition to being novel, to qualify for protection, "the difference between the subject matter sought to be patented and the prior art" must not "have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains". 35 U.S.C. § 103(a). A prior art reference can render a patent claim obvious on its own. *See SIBIA Neurosciences Corp.*, 225 F.3d at 1356. Obviousness is a question of law decided by the Court in light of the four factual factors outlined in *Graham v. John Deere Co.*, 383 U.S. 1, 17-18 (1966). *See generally Brown & Williamson Tobacco Corp. v. Phillip Morris Inc.*, 229 F.3d 1120, 1124 (Fed. Cir. 2000). The four *Graham* factors are: (a) the scope and content of the prior art; (b) differences between the prior art and the claimed invention; (c) the level of ordinary skill at the time the invention was made; and (d) objective evidence of non-obviousness. The ultimate determination of whether an invention is obvious is a legal conclusion based on the totality of the evidence. *See Brown & Williamson*, 229 F.3d at 1124. "Where . . . the content of the prior art, the scope of the patent claim, and the level of ordinary skill in the art are not in material dispute, and the obviousness of the claim is apparent in light of these factors, summary judgment is appropriate." *KSR Int'l Co.*, 550 U.S. at 427 (affirming grant of summary judgment on invalidity).

As explained in more detail below, application of the *Graham* factors compels the conclusion that the asserted claims were obvious over the prior art. There is little or no difference

between the teachings of Decasper98 by itself or in view of IBM96 and Nelson and the asserted claims. (*See* Parts IV and V, below.) This is true even if Implicit's level of ordinary skill in the art is used. (*See* Part III, below.) Finally, Implicit has not demonstrated that anything in the prior art taught away from the claimed invention, and none of the secondary considerations of non-obviousness negate the conclusion that the asserted claims were obvious. (*See* Part VI, below.)

II. IT IS UNDISPUTED THAT DECASPER98, IBM96, AND NELSON ARE PRIOR ART PUBLICATIONS UNDER § 102(b)

Printed publications from anywhere in the world are prior art under § 102(b) if they were published "more than one year prior to the date of the application for patent in the United States". 35 U.S.C. § 102(b). The date of the '163 application is December 29, 1999, and the '857 patent claims priority through that application. Thus, to qualify as a printed publication under § 102(b) in this case, a document would have to have been published prior to December 29, 1998.

Here, it is undisputed that Decasper98 was publicly available on or before September 10, 1998, that IBM96 was publicly available on or before May 1996, and that Nelson was publicly available on or before 1996 – *i.e.*, more than a year before December 29, 1999. (*See* Dixon Ex. 16 (Implicit's Responses to Juniper's First Set of RFAs) at 13; Dixon Ex. 14 (Implicit's Sixth Supplemental Response to Juniper's Second Set of Interrogatories (Nos. 11-19)) at 17.) As a result, Decasper98, IBM96, and Nelson qualify as prior art publications under § 102(b).[7]

III. THE LEVEL OF ORDINARY SKILL IN THE ART

Juniper and Implicit have proposed slightly different levels of ordinary skill in the art relevant as of December 1999, but these differences are immaterial to the resolution of this motion, as there is no indication that the differences would affect the manner in which any of these hypothetical "persons of ordinary skill" would have interpreted the prior art, and Implicit has presented no arguments to the contrary. (*See* Calvert Decl. ¶ 15.) Under either proposal, the hypothetical person would have been a person with at least a bachelor's degree in the relevant

[7] Moreover, because all of these publications trigger the statutory bar of § 102(b), the alleged date of invention is irrelevant to the invalidity analysis. *See* 35 U.S.C. § 102(b) (referring to prior art publication published "more than one year prior to *the date of the application for patent*") (emphasis added).

field of study – *e.g.*, computer science, computer engineering, electrical engineering, or a similar technical field – and a few years' experience – "at least two years' experience" for Implicit or "four years' experience" for Juniper. (*See* Calvert Ex. 2 (Calvert Report) ¶ 42; Dixon Ex. 17 (Nettles Rebuttal Report) ¶ 30.) Indeed, even if Implicit's hypothetical level of ordinary skill were used, the outcome of the obviousness inquiry would be the same. (*See* Calvert Decl. ¶ 15.)

IV. DECASPER98 ALONE RENDERS THE ASSERTED CLAIMS INVALID

A. Overview Of Decasper98.

Decasper98 describes a router system for processing messages or "flows" comprising multiple packets.[8] (*See* Dixon Ex. 1 at 1.) As a packet travels through the system, it encounters a sequence of "gates", each of which is associated with one or more "plugins". (*Id.* at 4; Dixon Ex. 17 ¶ 64 ("One of the flexibility aspects of Decasper is that, at a gate, different flows may have different plugins applied.").) Each "plugin" is a "dynamically loadable" software routine that implements a single function or transformation on a packet in a flow. (*See* Dixon Ex. 1 at 2, 4.) The Decasper98 plugins expect as input a piece of data (packet) having a specific structure or format, and output a piece of data having a specific, possibly different, format. (*Id.*)

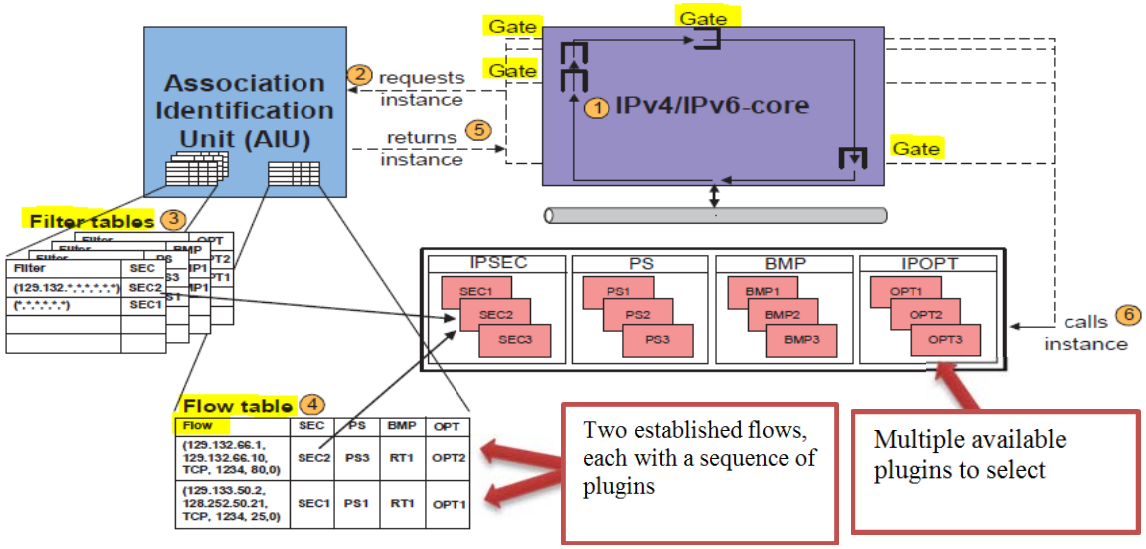
Decasper98 teaches that "[o]ne of the novel features of our design is the ability to bind different plugins to individual flows." (*Id.* at 1.) When the system encounters "the first packet of a new flow", "an entry" is created for it in a "flow table". (*Id.* at 5, 9.) The correct plugin for processing the packet at each particular gate is determined by consulting the independent "filter table" for the gate. (*Id.* at 5.) For example, a first filter table determines which plugin is selected for a first gate in the sequence, a second independent filter table determines which plugin is selected for a second gate in the sequence, and so on. (*See id.*) Of course, this ability to freely combine plugins means that there are an enormous number of possible sequences that might be applied to the first packet of a flow when it arrives, "even with very few installed filters". (*See id.*

[8] As explained in Decasper98, a "flow" may constitute "application streams" or "longer-lived packet streams with more packets per session than is common in today's environment," such as "real-time audio/video" stream. (Dixon Ex. 1 at 1-2.) "Flows" in Decasper98 are therefore synonymous with the Court's construction of the term "message": "a collection of data that is related in some way, such as a stream of video or an email message". (Dixon Ex. 30 at 13; *see also* Dixon Ex. 17 ¶ 61 (conceding that "[a] flow meets the Court's definition of 'message'").)

at 7.) These various possible sequences are not stored or enumerated anywhere in the system ahead of time. Instead, the sequence of plugins for a flow is generated after the first packet of a flow arrives, and then recorded in the flow table, so that when subsequent packets of the flow arrive, the correct plugins can be determined without having to consult the filter tables again, which is a computationally expensive operation. (*See id.* at 4-9.)

Decasper98 explicitly considers and rejects a "theoretically possible" alternative approach, which is to replace this system of multiple independent filters with "a single global filter table" that would contain all possible combinations of plugins set out in advance. (*Id.* at 7.) Under this alternative approach, only a single filter would apply to a particular flow, and that single filter would specify the entire sequence of components to be applied to it. (*See id.*) However, Decasper98 rejects this approach as "practically infeasible because the space requirements for the global table can, even with very few installed filters, increase very quickly (exponentially) to unacceptable levels." (*Id.*) In other words, Decasper98's multiple filter table approach implies so many potential valid sequences that it is impractical to even enumerate them all ahead of time in memory. Instead, Decasper98 adopts an algorithmic approach where the correct sequence is generated dynamically on the fly, after the first packet arrives.

For example, Figure 3 of Decasper98 depicts graphically how the system works:



(*Id.* at 5 (with annotations).) In this example, there are four gates – IPSEC, PS, BMP, and IPOPT – each with its own filter table on the left. (*Id.*) There are two flows, each with an entry in the

flow table. (*Id.*) As the first packet of the first flow encounters each gate, the correct plugin for that gate is determined and then recorded in the corresponding flow table entry. (*Id.*) For example, at an IP security gate ("SEC"/"IPSEC"), the plugin instance "SEC2" was chosen on the basis of the "SEC" filter table on the left, and this "SEC2" plugin was recorded in the first flow entry (in column "SEC"). (*Id.*) Similar processing was applied at the other gates, with the result that the plugin sequence SEC2, PS3, RT1, OPT2 was chosen for packets of the first flow, and recorded in its flow table entry. Similar processing was applied to the first packet of the second flow, but because its data was different, a different sequence of plugins was obtained from the series of four filter table lookups – *i.e.*, the sequence SEC1, PS1, RT1, OPT1. (*Id.*)[9]

B. Decasper98 Anticipates Or Renders Obvious The Asserted Claims Of The Patents Under §§ 102(b) Or 103.

As is evident from the discussion above, Decasper98 discloses an architecture that is extraordinarily similar to that claimed in the patents-in-suit, as even Implicit's expert (Dr. Nettles) admits. (*See* Dixon Ex. 18 (10/19/2012 Nettles Tr.) at 228:22-229:7) ("Well, Decasper has something happens on the first packet and a way of having that same set of things happen on subsequent packets and a way of remembering that. So that is a similar architecture.") Perhaps for this reason, Dr. Nettles disputes the presence of only ***three*** aspects of the asserted claims in Decasper98. (*See* Dixon Ex. 17 ¶¶ 67-75; Dixon Ex. 18 at 67:20-68:2 ("Well, I talk specifically about three aspects" of Decasper98.) Each of these disputed aspects is found in claim 1 of the '163 patent, and some of the same or similar limitations are found in other asserted claims. The language of claim 1 of the '163 patent is presented below with the disputed limitations enumerated and identified in **bolded, underlined** text:

Element[10]	**'163 patent, claim 1 (Dixon Ex. 5)**
1pre	1. A method in a computer system for processing a message having a sequence of packets, the method comprising

[9] More than just an academic exercise, the Decasper98 inventors built and tested a fully operable implementation of their invention. (*See, e.g.*, Dixon Ex. 1 at -2.)

[10] For the Court's convenience, Juniper adopts Implicit's own manner of referring to the various elements or limitations of the asserted claims. (*See, e.g.*, Dixon Ex. 19 (Nettles Infringement Rep.) at 10-16.)

1a	providing a plurality of components**, [i] <u>each component being a software routine for converting data with an input format into data with an output format</u>**;
1b	for the first packet of the message, **[ii] <u>dynamically identifying a non-predefined sequence of components for processing the packets of the message</u> [iii] <u>such that the output format of the components of the non-predefined sequence match the input format of the next component in the non-predefined sequence</u>**,
1c	wherein dynamically indentifying [*sic*] includes selecting individual components to create the non-predefined sequence of components after the first packet is received; and
1d	storing an indication of each of the identified components so that the non-predefined sequence does not need to be re-identified for subsequent packets of the message; and
1e	for each of a plurality of packets of the message in sequence, for each of a plurality of components in the identified non-predefined sequence, retrieving state information relating to performing the processing of the component with the previous packet of the message;
1f	performing the processing of the identified component with the packet and the retrieved state information; and
1g	storing state information relating to the processing of the component with packet [*sic*] for use when processing the next packet of the message.

As the PTO has already found in the ongoing *inter partes* reexamination proceedings, Decasper98 discloses each of these limitations, either standing alone and/or in combination with other references. The three disputed limitations are addressed in turn below.

1. *Decasper98 discloses and/or renders obvious element 1a of the '163 patent ("converting data with an input format into data with an output format").*

Decasper98 satisfies element 1a of the '163 patent, which recites "providing a plurality of components, each component being a software routine for converting data with an input format into data with an output format"**.** As described above (*see* Part IV(A)), Decasper98 describes providing a plurality of components (*i.e.*, plugins), each of which is a software routine ("plugins can be all software modules"). (Dixon Ex. 1 at 2, 4.) Each plugin provides one or more functions (*e.g.*, "authentication and/or encryption", "IPv6 options") for processing data in a packet with a specific structure or appearance, to produce data with a specific output structure or appearance. (*See id.* at 1-2, 4, 5; *see also* Calvert Ex. 2 ¶ 553.)

Dr. Nettles nevertheless contends that Decasper98 does not perform the "converting" aspect of element 1a, for two reasons.

First, Dr. Nettles argues that the Decasper98 plugins do not meet the "converting" requirement because they "***typically*** do not ***significantly*** alter the packets." (Dixon Ex. 17 ¶ 71.) But this argument (even if true) is irrelevant, as the claim language imposes no requirement that conversion be either "typical" or "significant". Thus, if a Decasper98 system were configured to (for example) only ***occasionally*** convert packets from a non-encrypted format to an encrypted format using a weak encryption algorithm, that would satisfy element 1a just as well as a system that ***always*** encrypted traffic using a more elaborate or "significant" algorithm.

Second, Dr. Nettles argues that the Decasper98 plugins do not convert data "with an input format into data with an output format" because each plugin allegedly takes "as input a packet in the IP format and has as output in the IP format". (*Id.*) In other words, Dr. Nettles asserts that Decasper98's plugins do not satisfy this claim element because the input format is supposedly the same as the output format. Implicit has taken the same position in its interrogatory responses and filings in the ongoing reexamination proceedings. (*See* Dixon Ex. 14 at 35 ("[T]he Decasper router only processes one data format (IP).) There is no notion of changing data format in this reference."); Dixon Ex. 29 (Implicit Office Action Response, '163 Patent Reexam) at 21 (arguing against the applicability of prior art including Decasper because packets "are all IP packets" and there is no evidence of components that change the format of the packet).)

Of course, the problem with Implicit's position requiring different input and output formats is that it runs directly contrary to (1) the Court's claim construction in this case and (2) the arguments that Implicit itself advanced at the claim construction phase. As the Court will recall, it was Juniper that argued during the claim construction phase that the terms "input format" and "output format" should be construed "so that the 'format' of the output data of a packet is '***different***' from the input". (Dixon Ex. 30 at 9.) But Implicit vigorously opposed this proposed construction and asked for a much broader interpretation, arguing that "***there is nothing in the specification that . . . requires that the output format of a component be different from the input format***." (Dixon Ex. 12 (Implicit's Reply Brief) at 11 (internal quotations omitted); *see also id.* at 11-12 (citing alleged embodiment in the specification for which "the output format will not be different from the input format"); Dixon Ex. 13 (1/19/2012 Hearing Tr.) at 92:21-93:8.)

The Court ultimately agreed with Implicit's earlier position that the "input format" did not have to be the different from the "output format", and construed the claims accordingly. (Dixon Ex. 30 at 9.) Now, having argued (successfully) that the asserted patent claims are broad enough to cover "single format" systems performing no conversion, Implicit cannot be heard to attempt to distinguish Decasper98 on this basis. *See Source Search Techs., LLC v. LendingTree, LLC*, 588 F.3d 1063, 1075 (Fed. Cir. 2009) (claims must be construed in the same way for both infringement and validity); *Liebel-Flarsheim Co. v. Medrad, Inc.*, 481 F. 3d 1371, 1380 (Fed. Cir. 2007) (affirming judgment of invalidity based on patentee's overbroad claim construction; "[t]he motto, 'beware of what one asks for,' might be applicable here.").[11]

Nevertheless, even if the Court were to revisit its claim construction for this term – or find that Implicit has now disclaimed "non-converting" components by virtue of its disclaimers to the PTO during reexamination – Decasper98 would still satisfy the "converting" requirement of element 1a. For example, Juniper's expert Dr. Calvert has explained that Decasper98 discloses a plugin capable of performing "encryption" as described in RFC 1825 (a standards-based security architecture), which would convert data with an input format (unencrypted data) into data with a different output format (encrypted data). (*See* Calvert Ex. 2 ¶ 553; Dixon Ex. 1 at 2, 4, 5.) There is no real dispute on this point; Implicit has admitted that "[e]ncryption/decryption" is an example of format conversion within the scope of claim 1 of the '163 patent. (*See* Hefazi Ex. 28 (Implicit's 6/7/2012 '163 Infringement Contentions) at 230-231.) Dr. Nettles likewise testified that "encryption or decryption" is an example of conversion per the claims. (*See* Dixon Ex. 20 at 191:11-192:16; Hefazi Ex. 26[12] at 12.) And Dr. Calvert cites yet additional examples of conversion in Decasper98 such as authentication or IP options processing. (Calvert Ex. 2 ¶ 553; Dixon Ex. 1 at 2, 12.)

[11] Notably, in performing his infringement analysis in this case, Dr. Nettles flipped back to Implicit's earlier claim construction position that "it wasn't a requirement that there be conversion". (Dixon Ex. 20 (9/10/2012 Nettles Tr.) at 191:11-192:16 ("it's clear in the patent [that] conversion isn't necessary").)

[12] "Hefazi Ex. __" refers to exhibits to the Declaration of Nima Hefazi in support of Juniper Networks, Inc.'s Motion for Summary Judgment of Non-Infringement, executed November 9, 2012.

Accordingly, Decasper98 clearly satisfies element 1a of the '163 patent and similar limitations in other claims.[13]

2. *Decasper98 discloses and/or renders obvious element 1b of the '163 patent ("dynamically identifying . . . components . . . such that the output format . . . . match[es] the input format").*

Decasper98 also satisfies element 1b of the '163 patent, which recites:

> for the first packet of the message,
>
> dynamically identifying a non-predefined sequence of components for processing the packets of the message such that the output format of the components of the non-predefined sequence match the input format of the next component in the non-predefined sequence,

Implicit nevertheless argues that Decasper98 fails to meet two aspects of this element, which are addressed in turn below.

(a) "Dynamically identifying"

Decasper98 clearly discloses the "dynamically identifying" limitation. As described above, when the first packet of a new flow arrives, Decasper98 performs a detailed series of operations to determine the correct sequence of plugins (or "components" in the language of the claims) to be applied to the flow. (*See* Dixon Ex. 1 at 5-6 ("The processing of the first packet of a new flow . . . involves *n* filter table lookups to create a single entry in the flow table for the new flow.").) This series of filter operations does not need to be repeated for subsequent packets of the flow, because the new "entry . . . in the flow table serves as a fast cache for future lookup of packets belonging to that flow," and the entry "stores pointers to the appropriate plugins". (*Id.* at 5.) Thus, it is clear that the sequence of plugins is "dynamically identif[ied]" as the individual

[13] Without offering any details, Implicit's purported expert asserts that "[s]imilar language/concepts is found in all of the Asserted Claims". (Dixon Ex. 17 ¶¶ 67, 70, 73.) Although the basis for this statement is not clear, it appears that Implicit's expert may be taking the view that Juniper must show some disclosure of format conversion in order to satisfy the "processing" limitation (which is in all the claims) for purposes of its invalidity analysis. If so, this position is again contrary to the Court's claim construction order, which found (at Implicit's urging) that conversion "is not a necessary limitation with respect to the processing of each packet". (Dixon Ex. 30 at 8-9.) As the Court will recall, Juniper had proposed that the term "processing" be construed to require that "the data in each of the packets of the message is 'converted' in some manner." (*See id.* at 8.) In any event, there is no genuine dispute that Decasper98 indeed discloses format conversion, as demonstrated above.

plugins in the sequence are not selected until after the first packet of the message.

In response, rather than attempting to show that the sequence of ***plugins*** is not dynamically identified, Dr. Nettles bizarrely points to certain aspects of the ***gates*** in Decasper98, which he argues are not dynamic but rather "fixed" or "predefined" at "compile time". (*See* Dixon Ex. 17 ¶¶ 63, 69); *see also* Dixon Ex. 18 at 200:9-208:16, 246:16-249:3.) This conflation of gates and plugins badly miscomprehends Decasper98. In Decasper98, a "gate is a point in the IP core where the flow of execution branches off to an instance of a plugin." (Dixon Ex. 1 at 4.) Plugins, on the other hand, are the "dynamically loadable kernel modules" that implement a single function or transformation on a packet. (*Id.*) Indeed, as Implicit recognizes, "[o]ne of the flexibility aspects of Decasper is that, at a gate, different flows may have different plugins applied." (Dixon Ex. 17 ¶ 64.) These "dynamically loadable" plugins are distinct from the gates. Thus, even if the gates in Decasper98 were predefined, that would in no way require that the plugins be predefined as well.

Indeed, the PTO has already directly considered – and rejected – Implicit's "predefined gates" argument for this element in the context of the '163 patent reexamination. (*See* Dixon Ex. 15 at 13 ("[A]lthough dynamic selection of a plugin may happen at a 'gate' (described as 'a point in the IP core where the flow of execution branches off to an instance of a plugin'), *the gate is not the plugin itself*.").) As the Examiner explains in addressing this limitation in the Action Closing Prosecution, the gates are "irrelevant" because it is clear that different sequences of individual ***plugins*** can be selected for different flows, after a first packet is received:

Flow table (4)

Flow	SEC	PS	BMP	OPT
(129.132.66.1, 129.132.66.10, TCP, 1234, 80,0)	SEC2	PS3	RT1	OPT2
(129.133.50.2, 128.252.50.21, TCP, 1234, 25,0)	SEC1	PS1	RT1	OPT1

The fact that Decasper98 can be viewed loosely as performing the same general type of processing at a gate (e.g., security processing of some kind at a security gate, packet scheduler processing of some kind at a packet scheduling gate) is irrelevant. It is the selection of individual components (e.g., SEC2 or SEC1 plugin, PS3 or PS 1 plugin) based on characteristics of the first packet received that matters for purposes of the claims.

(*Id.* at 14-15.) The Examiner therefore ultimately concludes in the Action Closing Prosecution that the Decasper98 approach thus "falls squarely within the scope of the claims" and indeed that the "selection from pools of different possible plugins is fundamental to [Decasper98's] design." (*Id.* at 15.[14])

Accordingly, Decasper98 clearly discloses the "dynamically identifying" aspect of element 1b of the '163 patent and similar limitations in other claims, including elements 15a, 35a, and 35b of the '163 patent and elements 1b, 4a, and 10b of the '857 patent.

(b) Matching compatibility of input and output formats[15]

Decasper98 also satisfies the third of the three aspects of the claims challenged in Dr. Nettles's report: "such that the output format of the components of the non-predefined sequence match the input format of the next component in the non-predefined sequence." The substance of this limitation is also captured in the Court's construction of the term "selecting individual components", which appears in every asserted claim: "selecting the individual software routines of the sequence so that the input and output formats of the software routines are compatible." (Dixon Ex. 30 at 11.[16]) The claimed concept centers on the common-sense proposition that, in order to have a properly functioning sequence of components in a system, each component needs to be compatible with its neighbors in that sequence.

The two Examiners in the respective reexamination proceedings for the '163 and '857 patents both found this limitation satisfied by Decasper98, although each reached this conclusion in a slightly different way. Examiner Whittington found an inherent disclosure in Decasper98 on this point, observing that the "output format of the components/plugins of the sequence would ***necessarily*** or ***inherently*** match the input format of the next component/plugin – ***otherwise the***

[14] *See also* Decasper98, Figure 2 (showing a pool of several possible components that may be identified for each gate: *e.g.*, several possible components for an IP Options gate, a Packet Scheduling gate, etc.).

[15] Juniper applies here, without prejudice, the broad understanding of this limitation that Implicit appears to have employed in connection with its infringement claims.

[16] *See* Dixon Ex. 23 at 1003:17-1004:14 (confirming the two formulations are "largely the same").

later plugin would not be able to perform its processing." (Dixon Ex. 22 (5/10/12 '857 Reexam Office Action) 8 (emphasis added); *Mehl/Biophile Int'l Corp.*, 192 F.3d at 1365 ("[I]f the prior art necessarily functions in accordance with, or includes, the claimed limitations, it anticipates.").) In turn, Examiner Ahmed concluded in his initial Office Action (and later again in an Action Closing Prosecution) that although Decasper98 did not "explicitly teach" this aspect of the claims, it was "certainly obvious":

> [I]t was well-known to those of ordinary skill in the art that certain operations on a packet must be performed in a certain order: e.g., if a packet is first converted into an encrypted format by a first component, a subsequent component would be unable to, e.g., process any IPv6 option headers in the packet, or to insert any new ones (because it was expecting to receive the packet in an unencrypted format). Thus, it was certainly obvious for one of ordinary skill in the art to arrange the sequence of components in a compatible manner, such that the output format of one matches the input format of the next.

(Dixon Ex. 15 at 103.) Examiner Whittington likewise adopted an obviousness rejection on similar grounds in the alternative. (*See* Dixon Ex. 22 at 10-11.) Juniper's expert Dr. Calvert has similarly opined that Decasper98 either discloses or renders obvious the input/output compatibility aspect of the claims. (*See* Calvert Ex. 2 ¶¶ 555, 585; *see also id.* ¶¶ 614, 618, 620, 748.)[17]

Dr. Nettles again provides just two arguments in response.[18] *First*, Dr. Nettles contends that "there is simply no reason in Decasper[98] to even contemplate matching the input and output formats" because Decasper98 is "a one format system". (*See* Dixon Ex. 17 ¶ 74.) As discussed above, this argument fails as it is contrary to the Court's claim construction and Implicit's prior argument that even systems performing no format conversion whatsoever (*i.e.*, a basic router)

[17] One of the co-authors of the Decasper98 reference, Dr. Bernhard Plattner, also submitted a declaration in connection with Juniper's submissions to the PTO in the reexamination proceedings, which similarly opined on the inherency and obviousness of this point. (*See* Dixon Ex. 21 (8/9/2012 Plattner Decl.) ¶ 9.) Dr. Calvert relied on this declaration in his report. (*See, e.g.*, Calvert Ex. 2 ¶ 74.)

[18] Notably, when asked at his deposition whether it would be obvious to "put together . . . software routines in such a way that the input and output formats match", Dr. Nettles responded that he did not have an opinion on that subject. (Dixon Ex. 18 at 277:9-278:3 ("Well, I don't know. I haven't really been asked to – to think about that specific question.").)

could still fall within the scope of the claims. (*See* Part III(B)(1), above.) In any event (and as also explained above), Juniper has shown that Decasper98 indeed discloses format conversion (*e.g.*, from an unencrypted format to encrypted format) and so it is incorrect to characterize it as a "one format system".

Second, and relatedly, Implicit suggests that by virtue of being a "one-format system", Decasper98 "actually teaches a system in which matching of input and output is not required and not desirable, exactly the opposite of the claim limitation". (Dixon Ex. 17 ¶ 75.) This is also incorrect, as there is no teaching in Decasper98 that mandates against matching input and output formats (and Dr. Nettles never points to any). "Where, as here, art is silent on the capabilities or function of any particular item, that is not teaching away from its use." *Ex Parte Mary Smith*, Appeal 2007-1925, at 25 (BPAI June 25, 2007); *see also Panoptx, Inc. v. Protective Optics, Inc.*, 2007 U.S. Dist. LEXIS 83462, at *24-25 (N.D. Cal. Nov. 9, 2007) (recognizing that "silen[ce] on the subject" and even a reference's "description of one structure does not teach away from an alternative structure"). Moreover, this type of "teaching away" argument has no legal bearing on an anticipation rejection based on inherency. *See In re Schreiber*, 128 F.3d 1478 (Fed. Cir. 1997) ("The question whether a reference is analogous art is irrelevant to whether that reference anticipates.").

Accordingly, Decasper98 clearly discloses element 1b of the '163 patent and similar limitations in other claims, including elements 1b, 4a, and 10b of the '857 patent.

3. *Decasper98 anticipates or renders obvious the other limitations of the asserted claims, although they are undisputed.*

In addition, for completeness, Juniper addresses and summarizes below each and every limitation of the asserted claims – including uncontested unique limitations of the asserted claims – which Decasper98 clearly anticipates or renders obvious (as found by the PTO).

Exemplary claim language	**'163**	**'857**	**Decasper98 disclosure**
"A method in a computer system for processing a message having a sequence of packets"	1pre, 15pre, 35pre	1pre, 4pre, 10pre	Decasper98 teaches "an extensible and modular software architecture for high-performance . . . routers" that "allows code modules called plugins to be dynamically loaded into the kernel and configured at run time". (*See, e.g.*,

Exemplary claim language	'163	'857	Decasper98 disclosure
			Dixon Ex. 1 at 11; Calvert Ex. 2 ¶¶ 552, 559, 563, 568, 574, 579, 583, 590, 594, 598, 604, 609; Dixon Ex. 17 at 71:19-23.)
"a plurality of components, each component being a software routine for converting data with an input format into data with an output format"	1a [1b, 15a, 35a]	[1b, 4a, 10b]	*See* Part IV(B)(1), above.
"dynamically identifying a non-predefined sequence of components . . . such that the output format of the components . . . match the input format of the next component . . ."	1b, 15a, 35a, 35b	1b, 4a, 10b	*See* Part IV(B)(2), above.
"selecting individual components to create the non-predefined sequence of components after the first packet is received"	1c, 15c, 35c	1c, 4b, 10c	Decasper98 determines the sequence of individual plugins for a flow by applying filters. (*See* Dixon Ex. 1 at 5-6; Calvert Ex. 2 ¶¶ 554-555, 560, 570, 575, 580, 585, 591, 600, 605.) These possible sequences are not stored or identified before the first packet of a message is received. (*See also* Part IV(B)(2), above.)
"storing an indication of . . . the identified components"	1d	1d, 4c, 10d	Decasper98 determines the sequence of individual plugins for a flow by applying filters and then stores pointers to the necessary plugins for a particular flow in a corresponding "flow table". (*See, e.g.*, Dixon Ex. 1 at 5-6; Calvert Ex. 2 ¶¶ 556, 571, 576, 580, 587, 601, 606).)
"for each of a plurality of packets . . . for each of a plurality of components . . . retrieving state information . . . performing the processing . . . with the . . . retrieved state information . . . storing state information . . . for use when processing the next packet . . ."	1e, 1f, 1g, 15d, 15e, 35d, 35e	1e, 1f, 4d, 4e, 10e, 10f	Decasper98 discloses the retrieval, use, and storage of state information on a component-by-component basis through a pair of pointers, including one that points to "private data for that plugin". For example, an encryption plugin or a statistics plugin would maintain various forms of state information – *e.g.*, counter-based values that are retrieved, used, and stored on a packet-by-packet basis. (*See, e.g.*, Dixon Ex. 1 at 9; Dixon Ex. 27 (RFC 1825) at 1, 3-5, 8-9; Dixon Ex. 28 (RFC 1883) at 1, 28-29; Calvert Ex. 2 ¶¶ 557, 561, 565, 572, 577, 588,

Exemplary claim language	'163	'857	Decasper98 disclosure
			592, 596, 602, 607; Dixon Ex. 17 at 83:9-84:7 ("I haven't disputed anything about the specific use of state in my report as best I remember.").)
"demultiplexing packets of a message"	15pre, 35pre	1f, 4e, 10f	Decasper98 receives a packet of a message, identifies a sequence of plugins, and then processes the message through the particular sequence of plugins. (*See, e.g.*, Dixon Ex. 1 at 1-2, 4-5; Dixon Ex. 4 at 2:61-64; Calvert Ex. 2 ¶¶ 559, 561, 565, 572, 577, 580).)
"different . . . sequences of components can be identified"	15b		Decasper98 applies a series of filters to each flow, wherein each filter may select a specific plugin component implementing a different policy. (*See, e.g.*, Dixon Ex. 1 at 5-7; Calvert Ex. 2 ¶ 560).)
"instructions for . . ."	35pre	10pre	Decasper98 discloses "router plugins" that are "software modules that are dynamically loaded into the kernel and are responsible for performing certain specific functions on specific network flows". (Dixon Ex. 1 at 2.) One of ordinary skill in the art would recognize that these software modules would be dynamically loaded from a "computer-readable medium", such as a hard disk in the device. (*See* Calvert Ex. 2 ¶ 563.)
"receiving a packet of the message and a data type of the message"		1a, 10a	As packets arrive in the system disclosed in Decasper98, they are classified into flows based upon the values of several headers fields including "protocol" (which is a data type), "source port", and "destination port". (*See, e.g.*, Dixon Ex. 1 at 3, 5; Calvert Ex. 2 ¶ 569, 580, 598, 610.)
"analyzing the plurality of headers . . ."		4a	Packets in Decasper98 are classified into flows based upon various packet header fields, including "<source address, destination address, protocol, source port, destination port, incoming interface>". (*See* Dixon Ex. 1 at 3, 5, 7; Calvert Ex. 2 ¶¶ 575, 605.)

In summary, because Juniper has established that there is no genuine dispute that Decasper discloses or renders obvious every element of the asserted claims of the patents-in-suit, the Court should grant summary judgment of invalidity for those claims.

V. DECASPER98 IN VIEW OF IBM96 AND NELSON RENDER THE ASSERTED CLAIMS OBVIOUS UNDER § 103

In addition to finding anticipation and obviousness rejections in light of Decasper alone, the PTO has also issued obviousness rejections involving Decasper98 as combined with two additional references: IBM96 and Nelson. (*See* Dixon Ex. 22 at 11-14.) These two additional references provide teachings pertinent to another of a plurality of plugins that one of skill in the art would have appreciated could be advantageously incorporated into the "extensible" plugin system of Decasper: a plugin for ***data compression and/or decompression***. (*Id.*; *see also* Calvert Ex. 2 ¶¶ 673-700.) IBM96 and Nelson teach that such a plugin could operate in accordance with the claims using (for example) an "LZ77" algorithm, which would involve retrieving state information (a compression "dictionary" or "sliding window" of data), processing the packet of the message using that state information, and then storing updated state information for use with the next packet of the message. (Calvert Ex. 2 ¶¶ 678, 691.)

The primary argument Dr. Nettles advances against this obviousness combination is that there would have purportedly been "no motivation to combine such different references/systems." (Dixon Ex. 17 ¶ 102.) Although he does not provide a detailed explanation for this argument, it appears to be based on the mistaken assumption that IBM96 is a primarily a reference regarding "compression, especially stateful compression". (*Id.*) Yet although IBM96 does address the topic of stateful compression, that is not, in fact, the primary focus of the book. IBM96 is entitled "Local Area Network Concepts and Products: Routers and Gateways," and teaches router platforms with various capabilities, including compression capabilities. (*See, e.g.*, Dixon Ex. 2 at 33, 95-96.) Thus, as Dr. Calvert has explained, "[i]t was obvious to supplement the teachings of Decasper98 with IBM96 because Decasper98 teaches an 'extensible' router software architecture, and IBM96 teaches features which would have been typical of routers of the time period." (Calvert Ex. 2 ¶ 674.) One of skill in the art would have further known to look to Nelson, in turn,

for information regarding the specific compression algorithms that could be used in an IBM-style router platform. (*Id.* ¶ 689.) Indeed, Nelson teaches lossless stateful ("adaptive") compression techniques that "adjust quickly to the data stream", such as the LZ77 algorithm described above. (*See* Dixon Ex. 3 at 9, 19, 21.) Thus, this combination of references would have been obvious and entirely appropriate, just as the PTO concluded. (Dixon Ex. 22 at 13 ("[i]t would have been obvious to provide the compression algorithms taught by IBM96 or one of the compression algorithms of Nelson as one of the plug-ins in the router of Decasper98."); *see also* Calvert Ex. 2 ¶¶ 674, 689.)

The only other contention Dr. Nettles advances as to this combination is that a compression plugin would not perform "format conversion." (Dixon Ex. 17 ¶ 110.) But in fact, both Implicit and its technical consultant Mr. Treskunov have already admitted that "compression/decompression" constitutes format conversion in accordance with the claimed invention. (*See* Hefazi Ex. 26 at IMP141522; Hefazi Ex. 28 at 230-231; *see also* Calvert Ex. 2 ¶ 676 ("A compression component which compressed some or all of an original packet would convert data . . . .").) Thus, Implicit's final argument fails, and the Court should grant summary judgment of invalidity for the asserted claims based on these obviousness grounds as well.

VI. OBJECTIVE EVIDENCE OF NON-OBVIOUSNESS

As described above, Decasper98 alone or in view of IBM96 and Nelson renders obvious the asserted claims. Thus, any "secondary considerations of non-obviousness" that Implicit may advance in opposition to this motion cannot nullify the clear teachings of Decasper98, IBM96, and Nelson of the limitations of the asserted claims.

In the ongoing reexamination proceedings, the PTO found that Implicit had "failed entirely to put forth any competent evidence of secondary indicia of non-obviousness." (Dixon Ex.15 at 10.) And indeed, Implicit has similarly failed to produce in this litigation any competent evidence of secondary considerations of non-obviousness that have any nexus to the claimed invention. *Demaco Corp. v. F. Von Langsdorff Licensing, Ltd.*, 851 F.2d 1387, 1392 (Fed. Cir. 1988) ("[T]he patentee in the first instance bears the burden of coming forward with evidence sufficient to constitute a prima facie case of the requisite nexus."). For example, in order to show

that it built an embodiment of its asserted claims, Implicit would need to show (at a minimum) that one of its alleged embodiments actually satisfied each and every element of at least one of those claims. Implicit not only failed to do this, but affirmatively refused to do so. (Dixon Ex. 31 (Implicit's Opposition to Mot. to Compel) at 1.) Moreover, Implicit has not shown any commercial success with its purported embodiments. Throughout this litigation Implicit claims to have entered into numerous sales and licensing contracts with many sophisticated companies, including Intel, Raytheon, and AMD. Notably, however, Implicit does not contend that any these contracts involved embodiments of the patents-in-suit, and indeed in the case of Raytheon, Mr. Balassanian testified that Implicit's work had nothing to do with the patents-in-suit. (Dixon Ex. 23 at 700:18-20.) Nor does Implicit's response identify any contracts with Raytheon or AMD. Implicit does refer to four specific contracts it entered into with Intel. But none of the products Implicit worked on for Intel was ever actually released to market; all were ultimately cancelled. Mr. Balassanian testified that Implicit never made any per-unit revenue on these cancelled projects. That is hardly an indication of commercial success. (*See also id.* at 448:3-5; 562-586; 581:20-582:8; 584:13-24; 565:19-23.)

Of course, evidence of "simultaneous invention" can be a secondary indicia tending to show obviousness. *Ecolochem, Inc. v. S. California Edison Co.*, 227 F.3d 1361, 1379 (Fed. Cir. 2000). As set forth in Dr. Calvert's expert report, there were numerous individuals, universities, and companies that were working in the same space as Implicit during the time frame of purported invention, including companies like Juniper/NetScreen, Cisco, and Checkpoint and researchers such as Decasper, Mosberger, and Pfeifer. (*See* Calvert Ex. 2 ¶¶ 77-90, 100-104, 175-335-341.) To the extent that Implicit claims to have "invented" anything or solved any purported problem, these groups and individuals (among others) invented the same thing and solved the same problem in the same way at around the same time (or earlier). That is the epitome of obviousness. This secondary consideration weighs heavily in favor of the obviousness of the asserted claims.

Accordingly, the Court should find, as a matter of law, that any secondary considerations here do not override the clear weight of evidence favoring a determination of obviousness in light of Decasper98 alone or Decasper98 in view of IBM96 and Nelson.

VII. IN THE ALTERNATIVE, PARTIAL SUMMARY JUDGMENT SHOULD BE GRANTED AS TO ANY LIMITATION ANTICIPATED OR RENDERED OBVIOUS BY DECASPER98 ALONE OR RENDERED OBVIOUS BY DECASPER98 IN VIEW OF IBM96 AND NELSON

Courts may grant summary judgment as to any issue (*i.e.*, any "part" or a claim or defense), even if summary judgment is denied as to the entire claim or defense. Rule 56 of the Federal Rules of Civil Procedure provides that "[a] party may move for summary judgment, identifying each claim or defense – *or the part of each claim or defense* – on which summary judgment is sought." Fed. R. Civ. P. 56(a) (emphasis added). Moreover, "[i]f the court does not grant all the relief requested by the motion, it may enter an order stating any material fact – including an item of damages or other relief – that is not genuinely in dispute and treating the fact as established in the case." Fed. R. Civ. P. 56(g). For example, in *Acco Brands, Inc. v. PC Guardian Anti-Theft Products, Inc.*, this Court granted summary judgment to the plaintiff as to one claim element but denied summary judgment as to the only remaining claim element in dispute, finding that factual questions remain that would need to be decided by a jury. 2008 WL 753899 (N.D. Cal. Mar. 18, 2008).

Here, too, if the Court finds that summary judgment is appropriate for some, but not all, of the limitations of the asserted claims because they are either anticipated or rendered obvious by the prior art, the Court should grant Juniper's motion as to those specific limitations. This would have the advantage of streamlining any trial in this case by focusing the jury's attention on matters of genuine dispute. Indeed, at the very least, the Court should grant Juniper's motion as to the limitations that Implicit does not contest are anticipated or rendered obvious by Decasper98. (*See* Part IV(B)(3), above.)

CONCLUSION

For the reasons set forth above, Juniper respectfully requests that claims 1, 15, and 35 of the '163 patent and claims 1, 4, and 10 of the '857 be held invalid as being anticipated or rendered obvious by Decasper98 alone or rendered obvious by Decasper98 in view of IBM96 and Nelson. All the asserted claims presently stand rejected in concurrent reexamination proceedings in the PTO on these grounds, and the same result should hold here.

Dated: November 12, 2012

Respectfully submitted,

IRELL & MANELLA LLP

By: */s/ Douglas J. Dixon*

David C. McPhie (dmcphie@irell.com)
Douglas J. Dixon (ddixon@irell.com)
Nima Hefazi (nhefazi@irell.com)
840 Newport Center Drive, Suite 400
Newport Beach, California 92660-6324
Telephone: (949) 760-0991
Facsimile: (949) 760-5200

Morgan Chu (mchu@irell.com)
Jonathan S. Kagan (jkagan@irell.com)
1800 Avenue of the Stars, Suite 900
Los Angeles, California 90067-4276
Telephone: (310) 277-1010
Facsimile: (310) 203-7199

Attorneys for Defendant
Juniper Networks, Inc.